



Europäisches Patentamt  
European Patent Office  
Office européen des brevets



(11)

**EP 1 193 616 A1**

(12)

**EUROPEAN PATENT APPLICATION**

(43) Date of publication:  
**03.04.2002 Bulletin 2002/14**

(51) Int Cl.7: **G06F 17/30**

(21) Application number: **00402692.8**

(22) Date of filing: **29.09.2000**

(84) Designated Contracting States:  
**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU  
MC NL PT SE**  
Designated Extension States:  
**AL LT LV MK RO SI**

• **Roy, Pierre**  
**75005 Paris (FR)**  
• **Cazaly, Daniel**  
**75017 Paris (FR)**

(71) Applicant: **Sony France S.A.**  
**92110 Clichy (FR)**

(74) Representative: **Bertrand, Didier et al**  
**c/o S.A. FEDIT-LORIOT & AUTRES**  
**CONSEILS EN PROPRIETE INDUSTRIELLE**  
**38, Avenue Hoche**  
**75008 Paris (FR)**

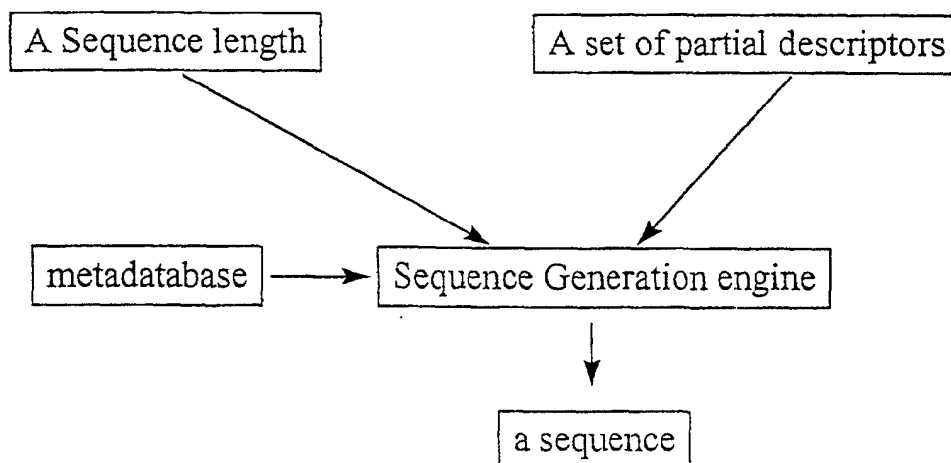
(72) Inventors:  
• **Pachet, François**  
**75005 Paris (FR)**

(54) **Fixed-length sequence generation of items out of a database using descriptors**

(57) The invention relates to a method of generating sequencing information representing a sequence of items selected in a database. Each of the items comprises a set of descriptors. The method is characterised by comprising the steps of: a) specifying a length of items and at least one of the descriptors; b) performing similarity relation techniques between the items; and c) generating a fixed-length sequence having a morphological continuity. Each of the items is represented by a

series of constraint variables having a domain in the database, and modelling each of the values of descriptors in a desired sequence as a constrained variable. The similarity relation performing step may comprise performing global similarity techniques by combining individual similarity measures on all of the descriptors. The sequence-generating step comprises transforming at least one of the values into unary constraints in terms of constraint satisfaction programming techniques.

**FIG. 3**



EP 1 193 616 A1

## Description

**[0001]** The present invention relates to a system that produces fixed-length sequences of items out of a database, such as music title programmes from a music catalogue. A sequence thus produced has to comply with partial information specified by a user, and to be "continuous" in terms of "morphologies" of the items generated in the sequence. This partial information may be, for example, a first title and a last title in the sequence of items (simple case), or part of "morphological" information on any particular item in the sequence. In the musical field, this information may be a musical descriptor such as genre, a type of rhythm, the tempo, etc. The "continuity" between the items is achieved through a similarity relationship. This relationship is defined as a combination of individual similarity measures described for each possible descriptor value.

**[0002]** Advances in networking and transmission of digital multimedia data has provided users with a huge number of information catalogues, such as music catalogues. These advances thus raise not only the problem of distribution, but also the problem of choosing desired information among huge catalogues.

**[0003]** Such new developments raise music selection problems which may depend on the aims of users or content providers. Although modelling a user's goal in accessing music is very complex, two basic elements, i.e. desire of repetition and desire of variation or surprise, can be identified.

**[0004]** The desire of repetition means that people want to listen to music they already know, or similar to what they already know. Sequences of repeating notes create expectations of the same notes to occur. On the other hand, the desire for variation or surprise is a key to understanding music at all levels of perception.

**[0005]** Of course, these two desires are contradictory, and the issue in music selection is precisely to find the right compromise: provide users with items they already know, or items they do not know but would probably like.

**[0006]** From the viewpoint of record companies, the goal of music delivery is to achieve a better exploitation of the catalogue. Indeed, record companies have problems with the exploitation of their catalogues using standard distribution schemes. For technical reasons, only a small part of a catalogue is actually "active", i.e. proposed to users, in the form of easily available products. More importantly, the analysis of music sales shows clearly decreases in the sales of albums, and short-term policies based on selling many copies of a limited number of items (hits) are no longer efficient. Additionally, the sales of general-purpose "samplers" (e.g. "Best of love songs") are no longer profitable, because users already have the hits, and do not want to buy CDs in which they like only a fraction of the titles. Instead of proposing a small number of hits to a large audience, a natural solution is to increase diversity, by proposing more customised albums to users.

**[0007]** In the present invention, the term "database" is used for designating any collection of data, e.g. covering both pre-stored data and dynamically stored data. The term "metabase" is used to describe a database containing descriptors of the items in the database. There are many situations in which it is necessary or desirable to create a sequence of items (e.g. music titles) from a collection of items for which data are available. It is also important that a created sequence is "coherent", i.e. there should exist a particular relationship between descriptors of the items which constitute a sequence. Typically, the descriptors of the items, components of the sequence, should not be too dissimilar, especially for successive items in the same sequence. A typical case where the problem supra arises is in the field of multimedia. A notable problem concerns an automatic generation of music programs, the latter being an example of temporal sequence. Here the term "program" is used not only to designate a sequence of musical pieces, but also, more generally, any temporal sequence of multimedia items, e.g. film clips, documentaries, texts.

**[0008]** A system producing "coherent" sequences of items in a particular order is disclosed in European patent application EP-A-0 961 209.

**[0009]** The items descriptors are stored in a metabase and consist of data pairs respectively consisting of a descriptor and a corresponding value. The problem of creating the desired sequence is treated as a "Constraint Satisfaction Programming (CSP)", also disclosed in the European patent application supra. The sequence to be obtained is specified by formulating a collection of constraints holding on items in the metabase. Each constraint describes a particular property of the sequence, and the sequence can be specified by any number of constraints.

**[0010]** The items in the metabase exhibit a particular generic format with associated taxonomies for at least some of the descriptors. Also, the constraints are specified out of a predetermined library of generic constraint classes which have been specially formulated. The special constraint classes allow the expression of desired properties of the target sequence, notably properties of similarity between groups of items, properties of dissimilarity and properties of cardinality. These constraint classes enable the properties of coherent sequences to be expressed in a particularly simple manner.

**[0011]** It is the combination of the use of a generic format for items in the data base and the special constraint classes which enables the use of a CSP solution technique to solve the combinatorial problem of building an ordered collection of elements satisfying a number of constraints.

**[0012]** It is an object of the present invention to provide a system which enables users to produce a fixed-length sequence of items out of a database by specifying only partial information. The main innovation of the invention is 1) the introduction of a special class of con-

straint, namely the global continuity constraint which allows to compute a "morphing" between two titles and 2) the possibility of specifying partial information about arbitrary titles in the sequence to be produced.

**[0013]** To this end, there is provided a method of generating sequencing information representing a sequence of items selected in a database, each of the items comprising a set of descriptors. The method comprises the steps of:

- a) specifying a length of the sequence and at least one of the descriptors;
- b) applying similarity relation techniques between the items; and
- c) generating a fixed-length sequence having a morphological continuity.

**[0014]** In the above method, each of the items may be represented by a series of constraint variables having a domain in the database.

**[0015]** Further, the above-mentioned similarity-relation applying step may comprise modelling each of the descriptors in a desired sequence as a constrained variable.

**[0016]** Further yet, the similarity-relation applying step may comprise applying a global similarity relation technique by combining individual similarity measures on all of the descriptors.

**[0017]** Typically, the similarity-relation applying step comprises providing mathematical similarity functions.

**[0018]** Suitably, the similarity-relation applying step comprises providing similarity relations defined by given thresholds.

**[0019]** In the above method, the sequence-generating step may comprise transforming the at least one of the values into unary constraints in terms of constraint satisfaction programming techniques.

**[0020]** Suitably, the above sequence-generating step further comprises subjecting the unary constraints to a processing of variables domain reduction.

**[0021]** In the above method, the descriptors are preferably expressed in terms of descriptor/value pairs respectively, and each of the values for the descriptor is selected from descriptor/value lists.

**[0022]** Further, each of the descriptors may be associated to a descriptor type.

**[0023]** Preferably, the above descriptor type comprises at least one type selected from the group consisting of Integer-Type, Taxonomy-Type and Discrete-Type.

**[0024]** In the above-described method, the step of specifying at least one of said values may comprise specifying a first title and a last title of the items in the sequence.

**[0025]** Further, the step of specifying at least one of the values may comprise specifying a morphological style of the items in the sequence.

**[0026]** In a typical case, the above database comprises musical pieces, and the values comprise titles, and

the titles form a music program.

**[0027]** The present invention further provides a system adapted to implement one of the above-described methods, which system comprises a general-purpose computer and a monitor for display of the generated information.

**[0028]** The invention also relates to a computer program product adapted to carry out one of the above-mentioned methods, when it is loaded into a general purpose computer.

**[0029]** The above and other objects, features and advantages of the present invention will be made apparent from the following description of the preferred embodiments, given as non-limiting examples, with reference to the accompanying drawings, in which:

Fig. 1 illustrates a taxonomy of musical styles, in which links indicate a similarity relation between styles. Here, "Jazz-Crooner" is represented as similar to "Soul-Blues";

Fig. 2 illustrates an example of music programs defined by descriptors;

Fig. 3 shows the general data flow according to the concept of the present invention; and

Fig. 4 illustrates an example of a user interface for specifying partial information on the descriptors for a sequence of length 8.

**[0030]** In the preferred embodiments, the invention is applied to the automatic composition of musical programmes, for example for radio, set-top boxes, etc..

**[0031]** The description of the preferred embodiments of the invention will begin with an explanation of the constitutive elements, on the basis of which the present invention is implemented.

**[0032]** The present invention therefore uses constraint satisfaction programming techniques already described in European patent application EP-A-0 961 209, whose corresponding part is herewith expressly incorporated by reference in its entirety.

**[0033]** In the technical field of the present invention, and more particularly in the musical field, applications targeted at non professionals have also been developed using "RecitalComposer", an embodiment of the previous patent application. "PathBuilder" is an application in which the user can specify a starting title and an ending title. The system contains hidden constraints on continuity of styles, and tempos are fixed. For instance: find a continuous path between Céline Dion's "All by myself", and Michael Jackson's "Beat it". Another similar application allows users to specify only the stylistic structure of the program. This may be used for instance for creating long programs for parties, in which the structure (e.g. begin with Pop, then Rock, then Slows, etc.) is known in advance.

**[0034]** Such an approach can be used to produce music programs in specific styles, by adding domain specific constraints. Other applications are envisaged for

set-top-box services and digital audio broadcasting.

**[0035]** As can be understood from the foregoing, "RecitalComposer" is an enabling technology for building high-level music delivery services. The system is based on the idea of creating explicit sequences of items, specified by their global properties, rather than on computing sets of items satisfying queries. One of its main advantages over other approaches is that it produces ready-for-use music programs which satisfy the goals of music selection: repetition, surprise, and exploitation of catalogues.

### Examples of the invention

**[0036]** The present invention is described hereinafter with reference to the sequences containing music titles. The invention relates to a method of specifying or describing some or the entirety of the descriptors of the titles in a sequence, thereby automatically generating distance measuring, and performing so called "morphing" between two or more items, e.g. music titles.

**[0037]** The present invention has the following features:

- 1) it allows the user to specify partial descriptions of titles (and not only a "specific description", i.e. entirely specified title); and
- 2) it allows the user to specify descriptions for "arbitrary" titles in the sequence (and not only the first or ending title).

#### 1. Metabase

**[0038]** The invention assumes the existence of a database of metadata, referred to as a metabase. The metabase of items, e.g. music titles, contains content information needed for specifying the constraints.

**[0039]** Each item is described in terms of descriptors which take their value in a predefined taxonomy. The descriptors are of two sorts: technical descriptors (descriptors) and content descriptors (values). Technical descriptors include the name of the title (e.g. name of a song), the name of the author (e.g. singer's name), the duration (e.g. "279 sec"), and the recording label (e.g. "Epic"). Content descriptors describe musical properties of individual titles. The descriptors may be the following: "style" (e.g. "Jazz Crooner"), "type of voice" (e.g. "muffled"), "music setup" (e.g. "instrumental"), "type of instruments" (e.g. "brass"), "tempo" (e.g. "slow-fast"), and other optional descriptors such as the "type of melody" (e.g. "consonant"), or the main "theme" of the lyrics (e.g. "love").

**[0040]** No assumptions are made as regards how the metabase is created. Some of the descriptors may be entered by hand, others extracted automatically, such as the tempo (see e.g. Scheirer, E.D., J. of the Acoustical Society of America, 103 (1), 588-601, 1998), or the rhythm structure (see previous patent application EP 00

401 915.4).

**[0041]** Although the invention is largely independent of the actual structure and content of the metadatabase, an example of such a metadatabase is given hereinafter. Typically, the descriptors include the following:

- Title name
- Author
- Style
- Tempo
- Energy
- VoiceType
- MainInstrument
- RhythmType

**[0042]** The possible values for each of these descriptors are taken from descriptor-value lists. Each descriptor is associated to a "Descriptor-Type". For instance, the Tempo descriptor is of Integer-Type (its value is an integer). The Style descriptor is of type "Taxonomy-Type". The MainInstrument descriptor is of type "DiscreteDescriptor", i.e. can take its value in a finite set of discrete values.

#### 2. Taxonomies of values and similarity relations

**[0043]** An important aspect of the metabase is that the values of content descriptors are linked to each other by similarity relations. These similarity relations are used for specifying constraints on the continuity of the sequence (e.g., the preceding example contains a constraint on the continuity of styles). More generally, the taxonomies on descriptor values establish links of partial similarity between items, according to a specific dimension of musical content.

**[0044]** For all descriptor types, there is supposed the existence of a similarity relation  $\text{similarity\_X}$ . This relation indicates whether a value for a given descriptor is "similar" to another value. For instance, the Style descriptor takes its value in a taxonomy of styles, in which the similarity relation is explicitly present (e.g.  $\text{style\_value} = \text{"Disco:US"}$  could be explicitly stated as similar to  $\text{style\_value} = \text{"Disco:Philadelphia"}$ ). Other descriptors can have mathematical similarity functions. For instance, the tempo descriptor ranges over integers, on which similarity relations are defined using thresholds:  $\text{similar-tempo}(a, b)$  if  $|b - a| < \text{threshold}$ .

#### 3. Input/Output

**[0045]** The present embodiment of the invention uses, as input,:

- (1) a sequence length  $n > 1$ , and
- (2) a limited number of descriptors (partial descriptors) for each item of the sequence.

**[0046]** It gives, as output, a sequence of length  $n$ ,

which satisfies a set of conditions defined below.

**[0047]** The "partial descriptors" are of the following form. For each title  $t_i$  of the sequence ( $1 \leq i \leq n$ , where  $n$  is the length of the sequence), any number of descriptors is given a possible value, including "non specified".

**[0048]** Fig. 4 shows an example of a user interface for specifying this partial information, for a sequence of length 8.

**[0049]** Once specified, the sequence is computed so that two conditions are satisfied:

- (i) titles of the sequence satisfy the corresponding partial specifications (when they exist, i.e. when they are different from "non specified");
- (ii) titles are linked to each other by a global similarity relation SIM, defined below.

#### 4. Algorithm

##### (A). Similarity relation

**[0050]** The algorithm uses a global similarity function SIM defined between two music titles. This function is a Boolean function (yields a yes/no answer). The function SIM can be defined in various ways. But in the present invention, all cases are defined from the individual similarity relations on each descriptor's "similarity-X" (see above).

**[0051]** The algorithm of the invention can in principle cope with any function SIM defined from similarity-X relations. In most cases though, the SIM function is defined as a logical combination of individual similarity-X relations. For instance, the SIM function can be defined as follows:

##### (B). Definition of the SIM function

**[0052]** The number of descriptors of  $t_1$  which are similar to the corresponding descriptors of  $t_2$  is less than 1.

**[0053]** In pseudo-code:

```

SIM( $t_1, t_2$ ) =
CPT = 0;
FOR I = 1 to Max-Descriptor
  if Similarity- $\neg$ ( $t_1, t_2$ ) = false, then CPT:= CPT +
  1;
end FOR
return CPT <= 1

```

##### (C). Description of the algorithm

##### (1). General aspect

**[0054]** The invention makes use of a constraint solver system, as in previous patent application EP 0 961 209.

**[0055]** The sequence generation problem is represented as a constraint satisfaction problem:

- the constrained variables are each title of the se-

quence  $t_i$ .

**[0056]** The domain of these variables is a database of titles.

- the constraints are the following:

i) Each "partial" specification given by the user is transformed into unary constraints, which are themselves transformed into variable domain reduction in a straightforward fashion. For instance, if the user wants the style of the third title to be "Jazz", then only items satisfying this constraint will be retained in the domain of  $t_3$ .

ii) The global similarity relation SIM is represented as a binary constraint established systematically between all pairs of contiguous title variables (i.e. between  $t_i$  and  $t_{i+1}$ , for  $i = 1$  to  $n - 1$ ).

**[0057]** A constraint satisfaction algorithm is applied using an arc-consistency technique for the similarity constraints, defined as follows:

**[0058]** The main specific aspect of this algorithm is the implementation of the filtering procedure for the binary SIM constraint. This constraint is implemented as follows.

##### (2) Description of the similarity constraint

##### (i) Notations

**[0059]** We present here the set of notations used in this section of the document.

**[0060]** The symbol  $\sim$  represents the similarity between descriptor values in their respective taxonomies

**[0061]** We use uppercase for variables, e.g. X, Y

**[0062]** We use lowercase for values, e.g. x, y

**[0063]** Dom(X) denotes the domain of variable X

**[0064]** C(X,Y) denotes a constraint involving two constrained variables X and Y

**[0065]** Constraint C(X,Y) is defined by a formula of satisfaction (intentional definition), e.g. if C(X,Y) is an equality constraint, it will be defined by:

$$C(X,Y)(x,y) = 1 \text{ iff } x = y$$

**[0066]** We systematically identify 1 with True and 0 with False

##### (ii) Preliminary

**[0067]** In our approach, we implement similarity constraints by means of simpler constraints: counting and descriptor constraints. Counting constraints are used to evaluate the number of differences between the descriptor values of two titles. Descriptor constraints are

used, in conjunction with descriptor variables, to represent descriptor values of titles as constrained variables. This is necessary to enable the definition of constraints over the descriptors themselves.

(a) Counting Constraints:  $CNT_R(X, Y, B)$

**[0068]** Given two constrained variables X and Y, given B a 0/1-constrained variable, and given a relation R defined over  $Dom(X) \times Dom(Y)$  -- the Cartesian product of the domains of X and Y, we define the constraint  $CNT_R(X, Y, B)$  (CNT stands for Counting constraint) by:

$CNT_R(X, Y, B)(x, y, b) = 1$  if  
 $(x R y)$  and  $b = 1$   
 or  $\text{not}(x R y)$  and  $b = 0$   
 $CNT_R(X, Y, B)(x, y, b) = 0$  otherwise

**[0069]** The filtering method for the CNT constraints consists of the following rules (every applicable rule is applied):

[Variable, Demon  $\Rightarrow$  Actions]  
 X, value(x)  $\Rightarrow$  {

If value(B) = 1, remove from  $Dom(Y)$  every y such that  $\text{not}(x R y)$   
 If value(B) = 0, remove from  $Dom(Y)$  every y such that  $x R y$   
 If for every y in  $Dom(Y)$ ,  $x R y$  holds, set the value of B to 1  
 If for every y in  $Dom(Y)$ ,  $x R y$  does not hold, set the value of B to 0}

Y, value(y)  $\Rightarrow$  {

If value(B) = 1, remove from  $Dom(X)$  every x such that  $\text{not}(x R y)$   
 If value(B) = 0, remove from  $Dom(X)$  every x such that  $x R y$   
 If for every x in  $Dom(X)$ ,  $x R y$  holds, set the value of B to 1  
 If for every x in  $Dom(X)$ ,  $x R y$  does not hold, set the value of B to 0}

B, value(b)  $\Rightarrow$  {

If b = 1

if value(X) = x, remove from  $Dom(Y)$  every y such that  $\text{not}(x R y)$   
 if value(Y) = y, remove from  $Dom(X)$  every x such that  $\text{not}(x R y)$

else (value(B) = 0)

if value(X) = x, remove from  $Dom(Y)$  every y such that  $x R y$   
 if value(Y) = y, remove from  $Dom(X)$  every x such that  $x R y$

**[0070]** Other demons do not trigger any action.

(b) Descriptor Constraints and Descriptor Variables

**[0071]** Given a constrained variable A and given a function f defined over  $Dom(A)$ , the descriptor variable f(A) is defined by its domain:

$Dom(f(A)) = f(Dom(A)) = \{f(a) \mid a \in Dom(A)\}$

**[0072]** Given two constrained variables A and B, and given a function f defined over  $Dom(A)$  and taking values in  $Dom(B)$ , we define the descriptor constraint  $ATTR_f(A, B)$  by:

$ATTR_f(A, B)(a, b) = 1$  if  $b = f(a)$

$ATTR_f(A, B)(a, b) = 0$  otherwise

**[0073]** The filtering of descriptor constraints is the following:

[Variable, Demon  $\Rightarrow$  Actions]

A, value(a)  $\Rightarrow$  set the value of B to f(a)  
 B, value(b)  $\Rightarrow$  remove every a from  $Dom(A)$  such that  $f(a) \neq b$   
 B, remove(b)  $\Rightarrow$  remove every a from  $Dom(A)$  such that  $f(a) = b$

**[0074]** Other demons (e.g., A, remove(a)) do not trigger any action.

(c) Similarity Constraints

**[0075]** Given two title variables A and B, and given a natural number N, the similarity constraint  $S_N(A, B)$  between A and B is defined by:

$S_N(A, B)(a, b) = 1$  if  $|\{i = 1 \dots P \mid \text{not}(a.i \sim b.i)\}| \leq N$

$S_N(A, B)(a, b) = 0$  otherwise

**[0076]** Which is equivalent to:

$S_N(A, B)(a, b) = 1$  if  $|\{i = 1 \dots P \mid a.i \sim b.i\}| \geq P - N$

$$S_N(A, B)(a, b) = 0 \text{ otherwise}$$

[0077] In these formulas, P represents the number of descriptors defined for a title, and "a.i" denotes the i<sup>th</sup> descriptor of title a.

[0078] To state a similarity constraint, we use descriptor variables (and descriptor constraints) to represent descriptor as constrained variables. We then define counting constraints to represent the number of similarities between descriptors of two titles as constrained variables. Then, we use a linear arithmetic constraint to limit the number of dissimilarities between two successive titles.

[0079] Technically, we define an additional descriptor variable for every descriptor of each title variable. Those descriptor variables are linked to the title variable they come from by an descriptor constraint. We then define a 0/1-variable for each descriptor variable. The 0/1-variable and the corresponding descriptor variable are linked together by a counting constraint. Optionally, we state a linear arithmetic constraint over the 0/1-variables which constrains the number of similarities between descriptors of the two title variables.

[0080] More precisely:

Let A and B be two title variables with P descriptors  
Let N be a natural number (between 0 and P)

[0081] We state the similarity constraint  $S_N(A, B)$  as follows:

[0082] For  $i = 1 \dots P$ , we define  $A_i$  (resp.  $B_i$ ) the descriptor variable of A (resp. B) corresponding to descriptor i; i.e. if style is the third descriptor,  $A_3$  is the variable whose domain is the set of styles of titles in the domain of A.

[0083] For  $i = 1, \dots, P$ , we state a constraint  $ATTR(A, A_i)$  (resp.  $ATTR(B, B_i)$ ) defined by  $ATTR(A, A_i)(a, b) = 1$  iff  $a.i = b$  (resp.  $ATTR(B, B_i)(a, b) = 1$  iff  $a.i = b$ )

[0084] For  $i = 1, \dots, P$ , we define a 0/1-variable  $C_i$

[0085] For  $i = 1, \dots, P$ , we state  $CNT_{\sim}(A_i, B_i, C_i)$ , the counting constraint for relation  $\sim$

[0086] We state a linear arithmetic constraint  $C1 + \dots + CP \geq P - N$

[0087] A similarity constraint on two title variables is therefore defined by:

[0088] 2.P descriptor variables (one for each title variable, and for each descriptor);

[0089] 2.P descriptor constraints linking every descriptor variable with the corresponding title variable;

[0090] P 0/1-variables, one for each pair of descriptor variables corresponding to the same descriptor;

[0091] P counting constraints, linking each pair of descriptor variables with one 0/1-variable;

[0092] One linear arithmetic constraint over the 0/1-variables.

[0093] The filtering of similarity constraints is achieved by the filtering of the different descriptor and

counting constraints defined... (properly speaking, the similarity constraint doesn't exist, it is a collection of additional variables linked together by descriptor and counting constraints)

(D). Applications of the invention:

[0094]

i) Sequences are generated by virtue of using the interface and algorithm described above.

ii) Iterative fixed-length sequences are generated, in which the user can iteratively apply the scheme to build sequences by refinement, e.g. by selecting, in a sequence computed by the system a title he/she does not want, and by relaunching the execution of the system.

## Claims

1. A method of generating sequencing information representing a sequence of items selected in a database, each of the items comprising a set of descriptors, **characterised in that** it comprises the steps of:

- a) specifying a length of said sequence and at least one of said descriptors;
- b) applying similarity relation techniques between said items; and
- c) generating a fixed-length sequence having a morphological continuity.

2. The method according to claim 1, wherein each of said items is represented by a series of constraint variables having a domain in the database.

3. The method according to claim 1 or 2, wherein said similarity-relation applying step comprises modelling each of said descriptors in a desired sequence as a constrained variable.

4. The method according to any one of claims 1 to 3, wherein said similarity-relation applying step comprises applying a global similarity relation technique by combining individual similarity measures on all of said descriptors.

5. The method according to any one of claims 1 to 4, wherein said similarity-relation applying step comprises providing mathematical similarity functions.

6. The method according to any one of claims 1 to 5, wherein said similarity-relation applying step comprises providing similarity relations defined by given thresholds.

7. The method according to any one of claims 1 to 6, wherein said sequence-generating step comprises transforming said at least one of said values into unary constraints in terms of constraint satisfaction programming techniques. 5
8. The method according to claim 7, wherein said sequence-generating step further comprises subjecting said unary constraints to a processing of variables domain reduction. 10
9. The method according to any one of claims 1 to 8, wherein said descriptors are expressed in terms of descriptor/value pairs respectively, and each of said values for said descriptor is selected from descriptor/value lists. 15
10. The method according to claim 9, wherein each of said descriptors is associated to a descriptor type. 20
11. The method according to claim 10, wherein said descriptor type comprises at least one type selected from the group consisting of Integer-Type, Taxonomy-Type and Discrete-Type. 25
12. The method according to any one of claims 1 to 11, wherein said step of specifying at least one of said values comprises specifying a first title and a last title of said items in said sequence. 30
13. The method according to any one of claims 1 to 12, wherein said step of specifying at least one of said values comprises specifying a morphological style of said items in said sequence. 35
14. The method according to any one of claims 1 to 13, wherein said database comprises musical pieces. 40
15. The method according to any one of claims 1 to 14, wherein said values comprise titles, and said titles form a music program. 45
16. A system adapted to implement the method of any one of claims 1 to 15, comprising a general-purpose computer and a monitor for display of the generated information. 50
17. A computer program product adapted to carry out the method of any one of claims 1 to 15, when loaded into a general purpose computer. 55



FIG. 1

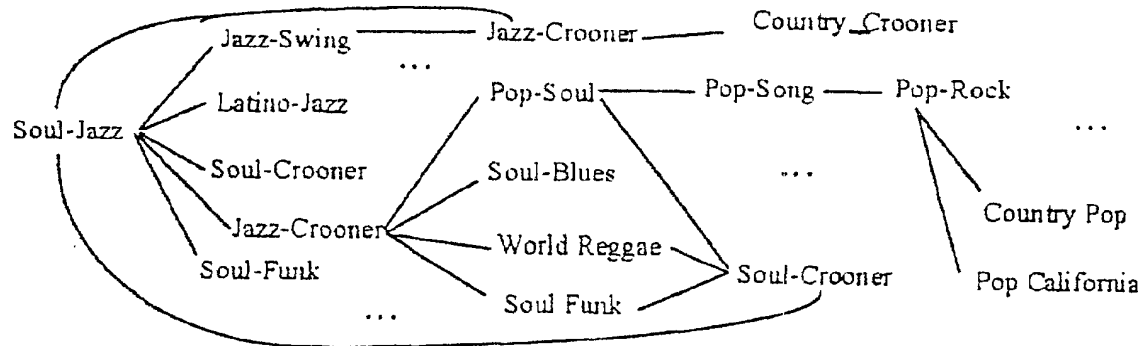
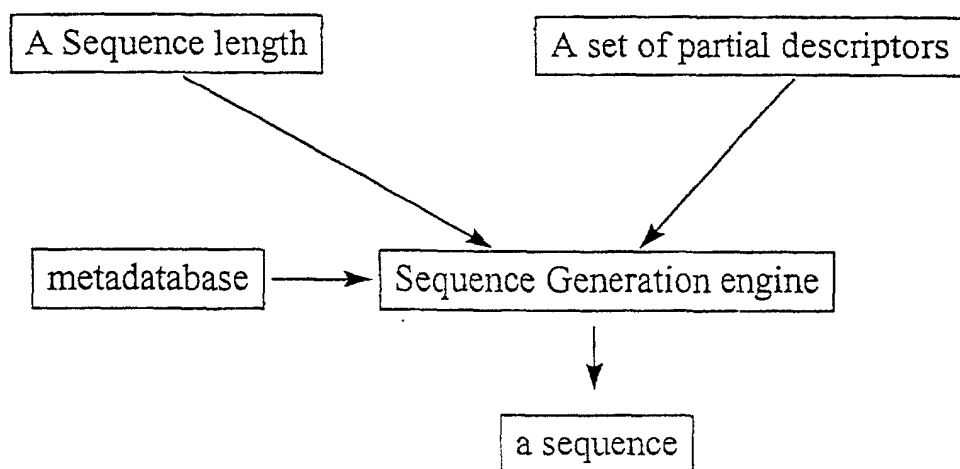


FIG. 2

3	sunrise	Atkins Chet instrumental	Jazz Calif Instrumental	250s jazz guitar	slow fast strings
21	surrounded	Kreviazuk Chant powerful	Pop Calif Woman	238s piano	slow fast strings
6	Still is still moving to	Nelson Willie nasal	Country Calif Man	210s calif guitar	fast calif guitar
9	Not a moment too soon	Mac Graw Tim hoarse	Country Calif Man	222s calif guitar	slow fast piano
10	Lovin' all night	Crowell Rodney normal	Country Pop Man	227s calif guitar	fast brass
11	Hard way (the)	Carpenter Mary normal	Country Pop Woman	262s calif guitar	slow fast piano
17	Point of rescue (the)	Ketchum Hal normal	Country Calif Man	265s calif guitar	fast calif guitar
50	At seventeen	Ian Janis soft	Pop Folk Woman	281s acoustic guitar	slow fast brass
27	Dream on	Labounty Bill broken	Pop Calif Man	298s keyboard	slow fast brass
106	Another time another plac	Steely Dan instrumental	Jazz Calif Instrumental	245s piano	fast slow keyboard
112	Learn to love you	Connick Harry J muffled	Jazz Crooner Man	279s brass	slow fast strings
137	Heart of my heart	Elgart Les instrumental	Jazz Swing Instrumental	151s double bass	slow fast brass

**FIG. 3****FIG. 4**

Title (code)	Europa							Just the way
Author	Santana							Billy Joel
Genre			Blues					
Instrument								piano
Voice								
Tempo					120			
Sequence length		8						
GO								



European Patent  
Office

# EUROPEAN SEARCH REPORT

Application Number  
EP 00 40 2692

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	PACHET F ET AL: "A combinatorial approach to content-based music selection" PROCEEDINGS IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS, PROCEEDINGS OF ICMCS99: IEEE MULTIMEDIA SYSTEMS '99: INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS, FLORENCE, ITALY, 7-11 JUNE 1999, pages 457-462 vol.1, XP002165216 1999, Los Alamitos, CA, USA, IEEE Comput. Soc, USA ISBN: 0-7695-0253-9	1,2,4-7, 12-17	G06F17/30
Y	* abstract * * page 458, right-hand column, line 27 - page 460, right-hand column, line 40 * * page 462, left-hand column, line 9 - page 462, left-hand column, line 52 * ---	3,9-11	
Y	PHILIPPE P: "Low-level musical descriptors for MPEG-7" SIGNAL PROCESSING. IMAGE COMMUNICATION, ELSEVIER SCIENCE PUBLISHERS, AMSTERDAM, NL, vol. 16, no. 1-2, September 2000 (2000-09), pages 181-191, XP004216275 ISSN: 0923-5965	3,9-11	TECHNICAL FIELDS SEARCHED (Int.Cl.7)
A	* abstract * * page 181, left-hand column, line 15 - page 181, right-hand column, line 19 * * page 182, left-hand column, line 19 - page 183, right-hand column, line 19 * --- -/--	1,16,17	G06F
The present search report has been drawn up for all claims			
Place of search <b>THE HAGUE</b>		Date of completion of the search <b>12 April 2001</b>	Examiner <b>Boyadzhiev, Y</b>
<p>CATEGORY OF CITED DOCUMENTS</p> <p>X: particularly relevant if taken alone Y: particularly relevant if combined with another document of the same category A: technological background O: non-written disclosure P: intermediate document</p> <p>T: theory or principle underlying the invention E: earlier patent document, but published on, or after the filing date D: document cited in the application L: document cited for other reasons &amp;: member of the same patent family, corresponding document</p>			

EPO FORM 1503 03/02 (P04001)



European Patent  
Office

# EUROPEAN SEARCH REPORT

Application Number  
EP 00 40 2692

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	PACHET F ET AL: "A combinatorial approach to content-based music selection" IEEE MULTIMEDIA, JAN.-MARCH 2000, IEEE, USA, vol. 7, no. 1, pages 44-51, XP002165217 ISSN: 1070-986X * abstract * * page 45, right-hand column, line 30 - page 46, left-hand column, line 12 * * page 46, right-hand column, line 5 - page 48, left-hand column, line 37 * * page 50, right-hand column, line 20 - page 50, right-hand column, line 43 * ----	1-17	
A,D	EP 0 961 209 A (SONY FRANCE S A) 1 December 1999 (1999-12-01) * abstract * * page 2, line 1 - page 3, line 26 * * page 4, line 1 - page 5, line 2 * * page 9, line 18 - page 9, line 52 * ----	1-17	
A	ROSENFELD P L: "HOME (OMPUTER) TERMINAL MUSICAL PROGRAM SELECTION" IBM TECHNICAL DISCLOSURE BULLETIN,US,IBM CORP. NEW YORK, vol. 23, no. 7B, 1 December 1980 (1980-12-01), pages 3440-3441, XP000706866 ISSN: 0018-8689 * the whole document * -----	1,16,17	TECHNICAL FIELDS SEARCHED (Int.Cl.7)
The present search report has been drawn up for all claims			
Place of search <b>THE HAGUE</b>		Date of completion of the search <b>12 April 2001</b>	Examiner <b>Boyadzhiev, Y</b>
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

EPC FORM 1503 03/92 (P04C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT  
ON EUROPEAN PATENT APPLICATION NO.**

EP 00 40 2692

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on  
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

12-04-2001

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0961209 A	01-12-1999	JP 2000067033 A	03-03-2000

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82